

A Grid-Based Algorithm in Conjunction with a Gaussian-Based Model of Atoms for Describing Molecular Geometry

Arghya Chakravorty ^[a] Emilio Gallicchio ^{*,[b]} and Emil Alexov ^{*,[a]}

A novel grid-based method is presented, which in conjunction with a smooth Gaussian-based model of atoms, is used to compute molecular volume (MV) and surface area (MSA). The MV and MSA are essential for computing nonpolar component of free energies. The objective of our grid-based approach is to identify solute atom pairs that share overlapping volumes in space. Once completed, this information is used to construct a rooted tree using depth-first method to yield the final volume and SA by using the formulations of the Gaussian model described by Grant and Pickup (*J. Phys Chem*, 1995, **99**, 3503). The method is designed to function uninterruptedly with the grid-based finite-difference method implemented in Delphi, a popular and open-source package used for solving the

Poisson–Boltzmann equation (PBE). We demonstrate the time efficacy of the method while also validating its performance in terms of the effect of grid-resolution, positioning of the solute within the grid-map and accuracy in identification of overlapping atom pairs. We also explore and discuss different aspects of the Gaussian model with key emphasis on its physical meaningfulness. This development and its future release with the Delphi package are intended to provide a physically meaningful, fast, robust and comprehensive tool for MM/PBSA based free energy calculations. © 2019 Wiley Periodicals, Inc.

DOI: 10.1002/jcc.25786

Introduction

Molecular geometry, best described by using the molecular volume (MV) or surface area (MSA) or both of a molecule, has served as a fundamental factor in modeling the nonpolar properties of biological macromolecules. The prominent role of nonpolar interactions in processes like the formation of protein aggregates in solvent, protein–drug binding, and membrane formation is well documented.^[1–3] Furthermore, binding free energy changes occurring due to mutations in proteins also conceive the important role played by the change in the surface area (SA) of the mutant sites in predicting the pathogenicity of the mutation.^[4,5] Besides binding, studies on folding and unfolding of proteins have signified the role of MV and MSA.^[6] From a geometrical perspective, these quantities help differentiate the level of packing of native from nonnative and unfolded states of proteins. From a thermodynamic perspective, changes in volume and SA signify the effect of pressure on protein folding in isothermal conditions, which is essentially the condition inside a biological cell. In addition, pressure-induced unfolding or denaturation and the associated volume changes of a protein have also been shown to have a significant dependence on the volumes of the internal cavities in its native structure.^[7,8]

To computationally determine the free energy of various macromolecular processes, the free energy is typically divided into its enthalpy part and its entropy part and the enthalpy part is further split into an intermolecular gas phase mechanical energy term, a polar energy term and a nonpolar energy term. Within the framework of implicit solvent models, the polar and the nonpolar components are computed using various models that are based on rational approximations of the underlying physics. These methods are widely popular and are integral

parts of protocols, such as MM/PBSA^[9] and MM/GBSA.^[10] The standard protocol is based on post-processing the configurations sampled by a molecular dynamics (MD) or Monte Carlo (MC) simulation in explicit water wherein the energy of individual configurations is computed and used to deliver the average free energy. For each configuration, the vacuum phase molecular-mechanical (MM) energy is computed using the force-field Hamiltonian and the associated atomic parameters. The polar component, which can be thought of as the work done to “turn on” the interactions of the solute charges with the solvent in a multidielectric media, is computed either via the Poisson–Boltzmann (PB) approach by using one of the several PB-solvers like Delphi,^[11] AMBER/PBSA,^[12] MIBPB,^[13] APBS,^[14] ZAP,^[15] ITPACT^[16] or the generalized born (GB) approach by using the models of Onufriev–Bashford–Case,^[17] Still,^[18] Hawkins–Cramer–Truhlar^[19] and many others. Finally, the nonpolar component, which can be thought of as the energy required to create a cavity in the bulk of the solvent large enough to accommodate the solute in question, can also be computed using various models, at the core of which lies the assumption that the nonpolar energy is related to the solute’s volume or SA or both.

However, the use of different models and their inclusion in the protocol for computing free energy can largely depend on

[a] A. Chakravorty, E. Alexov

Department of Physics and Astronomy, Clemson University, Clemson, South Carolina 29634

E-mail: ealexov@clemson.edu

[b] E. Gallicchio

Department of Chemistry, CUNY Brooklyn College, Brooklyn, New York
E-mail: egallicchio@brooklyn.cuny.edu

Contract Grant sponsor: NIH; Contract Grant number: R01GM093937

© 2019 Wiley Periodicals, Inc.

one's understanding of the underlying physics and one's expectations from these computations. As far as the nonpolar component of the free energies is concerned, which is also the main focus of this work, there are several different models that use MV and MSA to best rationalize their understanding for the underlying physical processes. Some empirical models assume a linear relationship between the nonpolar energy and the MSA^[20–26] while others also include the MV.^[27–33] By identifying some limitations of these linear models in describing the physical reality,^[34–36] recent models have suggested that in addition to the linear cavity term, an attractive van der Waals (vdW) term^[28,29,31,32,37–40] is also required to determine the total nonpolar contribution to the free energy. Key variations among these different models originate from their definition of protein volume and SA. Most models use the solvent-accessible surface area (SASA)^[23,34,41–44] of the proteins to quantify the size of the cavity while some also justify the use of van der Waals surface area (vdWSA).^[37–39,45] As for the volume, some use van der Waals volume (vdWV),^[37,39] others the solvent-accessible volume (SAV).^[31,33] In addition, these models may also differ in the method they use to represent individual solute atoms, for example, as classical hard-spheres that draws a strict boundary between the solvent and solute regions or as regions occupied by a smooth volume density (expressed as Gaussians) which promotes a strict-surface-free approach^[46,47] of describing solvated systems.

This variety in nonpolar energy models has called for different computational methods of computing volumes and SA. Besides differing on the model of atoms, these computational methods can also be distinguished in terms of the algorithm they use for identifying atomic overlaps^[48–50] or that for delineating surfaces of contact of the probe and the solute atoms.^[51–55] The use of one model over the other is certainly influenced by the time-efficacy and robustness besides the all-important physical meaningfulness. But as the number of structures in the protein data bank (PDB) grows and genomic expansion studies are being undertaken widely, researchers are using a large number of structures in their studies and are sampling larger configurational spaces for a better and holistic understanding of biomolecular processes. As a result, the time-efficacy of a computational method has become a significant factor in influencing its choice over others.

The design of such a fast and accurate method is the main goal of this work. The method we present here combines a novel grid-based approach of identifying overlapping atoms and the analytical approach of computing MVs and SA using a Gaussian-based description of atoms.^[49] The primary motivation is to integrate a method of computing MV and MSA, and therefore, a method of computing nonpolar energy terms, into the popular PB-solver Delphi.^[11] The use of a smooth Gaussian-based model will make this merger consistent with its smooth Gaussian-based approach of representing the dielectric distribution of solvated biomolecular systems.^[15,47] This integration is expected to provide a comprehensive platform for computing the free energy using a single package and thereby, offer a wide range of users a convenient way of analyzing and evaluating the energy of

system configurations sampled from large-scale simulations using the MM/PBSA protocol.

The novel grid-based algorithm is designed to identify pairs of solute atoms that overlap in space by simultaneously using the robust grid-based finite-difference method that Delphi uses to solve the Poisson–Boltzmann equation (PBE). By doing so, we show that little to no additional time is spent in identifying overlapping atom pairs. After the pairs have been identified, a depth-first tree-based algorithm, used by the popular AGBNP^[37] package, is used to compute the volumes and SAs.

The layout of this paper is as follows. First, the design of the grid-based search for overlapping atom pairs and its implementation in Delphi is demonstrated. Then the method is validated and benchmarked. Thereafter, we present a comparison of the Gaussian-based approach with respect to the standard hard-sphere model to highlight its numerical accuracy and physical appeal; the latter being shown in terms of the profile of volume and SA changes as a function of separation of the monomer units of Barnase–Barstar complex. We also present some aspects of the Gaussian model that have not been emphasized previously while reviewing and discussing some of the properties that are well known. We also review a previously reported modification to the Gaussian model, discuss its physical implications and highlight limitations to its applicability.

Theory and Implementation

The Gaussian model of computing MV and MSA

The Gaussian model prescribed by Grant and Pickup^[49] is presented here. We have adopted some new conventions and symbolisms to describe the model which shall be clarified as we explain it.

An atom “*i*” with vdW radius R_i and coordinate \vec{r}_i is described in a Gaussian representation via a density function given by.

$$g_i = p_i e^{-\alpha_i(\vec{r}-\vec{r}_i)} \quad (1)$$

with argument α_i of Gaussian exponent function expressed as

$$\alpha_i = \frac{\kappa}{R_i^2} \quad (2)$$

Here κ is a dimensionless parameter (see below), and height factor

$$p_i = \frac{4\pi}{3} \left(\frac{\kappa}{\pi}\right)^{\frac{3}{2}} \quad (3)$$

such that the volume, obtained from the volume integral of this density function, equals the hard-sphere volume ($V_i = \frac{4}{3}\pi R_i^3$) of the atom

The product, $g_{ij} = g_i g_j$, of the Gaussian density functions for atoms *i* and *j* which describes the volume of overlap between the two, is itself a Gaussian density function centered at

$$\vec{r}_{ij} = \frac{\alpha_i \vec{r}_i + \alpha_j \vec{r}_j}{\alpha_{ij}} \quad (4)$$

and Gaussian exponent

$$\alpha_{ij} = \alpha_i + \alpha_j$$

Correspondingly, in the Gaussian formalism the overlap volume, V_{ij} , of two atoms is given by the volume integral of their product density.

$$V_{ij} = \int_V dV g_{ij} = p_{ij} e^{-\left(\frac{\alpha_{ij}}{a_{ij}}\right)} \left(\frac{\pi}{\alpha_{ij}}\right)^{\frac{3}{2}} \quad (5)$$

where

$$p_{ij} = p_i p_j$$

and

$$A_{ij} = \alpha_i \alpha_j |\vec{r}_i - \vec{r}_j|^2 \quad (6)$$

This strategy can be extended recursively to obtain analytic expressions of the Gaussian overlap volumes of any order. For instance, the third order overlap volume of atoms i , j , and t is expressed as

$$V_{ijt} = \int_V dV g_{ijt} = p_{ijt} e^{-\left(\frac{\sum_{m,n=\{i,j,t\}} \alpha_{mn} A_{mn}}{a_{ijt}}\right)} \left(\frac{\pi}{\alpha_{ijt}}\right)^{\frac{3}{2}} \quad (7)$$

where

$$p_{ijt} = p_{ij} p_t$$

$$\alpha_{ijt} = \alpha_{ij} + \alpha_t$$

and

$$\vec{r}_{ijt} = \frac{\alpha_{ij} \vec{r}_{ij} + \alpha_t \vec{r}_t}{\alpha_{ijt}} \quad (8)$$

To compute the MV, overlap volumes are added to or subtracted from the arithmetic sum of the hard-sphere volumes of all the atoms, based on their order (inclusion–exclusion formula). The alternative inclusion and exclusion ensure that there is no redundancy in the contribution by a certain overlap region to the total volume.

$$V_{\text{molecule}} = \sum_i \frac{4}{3} \pi R_i^3 - \left(\sum_{i < j} V_{ij}^{\text{overlap}} - \sum_{i < j < t} V_{ijt}^{\text{overlap}} + \sum_{i < j < t < s} V_{ijts}^{\text{overlap}} + \dots \right) \quad (9)$$

The terms in the parenthesis in the right-hand side comprise the total overlap volume. Note that they occur with alternating signs of the form $(-1)^n$ where n is the order of the overlap.

The surface area SA_i of atom " i ", is defined as the derivative of the MV with respect to the radius of that atom. The total SA of the molecule is obtained from eq. (9) as the sum of the individual atomic SAs as

$$SA_{\text{molecule}} = \sum_i SA_i = \sum_i \left(\frac{\partial V_i}{\partial R_i} - \sum_j \frac{\partial V_{ij}}{\partial R_i} + \sum_{j,t} \frac{\partial V_{ijt}}{\partial R_i} - \sum_{j,t,s} \frac{\partial V_{ijts}}{\partial R_i} + \dots \right) \quad (10)$$

In the context of the Gaussian model, overlap volumes and their derivatives are available in analytic form. For a generic overlap term of order n , the derivative with respect to the radius of atom i is given by:

$$\frac{\partial V_{ij\dots n}}{\partial R_i} = \frac{\partial V_{ij\dots n}}{\partial \alpha_i} \left(\frac{\partial \alpha_i}{\partial R_i} \right) = \frac{2\kappa_i}{R_i^3} \left[\frac{3}{2\alpha_{ij\dots n}} + |\vec{r}_i - \vec{r}_{ij\dots n}|^2 \right] V_{ij\dots n} \quad (11)$$

A grid-based method of identifying overlapping atom pairs: algorithm

The above mathematical description of the model emphasizes on the importance of the overlapping volume and SA terms to these calculations. These terms are contributed by atoms that share a region, which means that each atom has its own set of neighboring atoms that affect its volume/SA. Typically, such pairs of atoms are found using a distance criterion, wherein two atoms, i and j , are said to be overlapping if:

$$|\vec{r}_i - \vec{r}_j| \leq R_i + R_j + \epsilon \quad (12)$$

where R represents their respective radius and \vec{r} designates their center coordinates, such as those provided in a PDB file. ϵ , typically, has a small value that provides allowance for those pairs of atoms which would not overlap were they to be described as classical hard-spheres. Finding out this pair-list, also known as neighbors list, therefore, requires $O(N^2)$ operations, in theory. Algorithms like cell-linked list,^[56] domain-decomposition method,^[57] Verlet list^[58] and others^[59–61] were contrived solely to cut down on the computation time and are mainly incorporated with MD simulation packages.

In our approach, we make use of the network of grids constructed by Delphi in order to solve the PBE using finite difference method.^[11,62] The neighbor list of the atoms is computed in this symmetrical 3D mesh of grids (also called box) on which the molecule in question is projected into. The box is large enough to accommodate the molecule fully and have an additional space around it to account for the solvent phase. Based on the number of grids per Å (a.k.a resolution or "scale"), the finesse of the 3D mesh can be manipulated. At its core, these grid points serve as points in space where the electrostatic potential and other quantities like electrolyte concentration are determined.

The first step of Delphi's algorithm is to determine the dielectric distribution of the system contained in the box. With the information of the coordinate of the atoms and their radii, grid points are surveyed and based on its distance from the center,

a dielectric value is assigned. As evaluating all the grid points can be extremely expensive, only a cubic region around the atom in question, large enough to accommodate its spherical volume is scanned.^[11] Consecutive atoms are projected onto the grid points and a 3D dielectric distribution map is constructed. It is at this step that the neighbor list of atoms is generated. As consecutive atoms are placed onto the grid, computation of neighbor list runs in parallel which uses the following criteria to identify neighbors: *two atoms are considered as neighbors if the local cubic box around them share at least one grid point*. If the boxes are larger, more neighbors will be identified and vice versa. However, overestimation of the number of neighbors will not necessarily overestimate the volume. It will simply increase the computation time.

At this point, we would like to alert the readers that though the Gaussian model describes atoms as densities, our grid-based approach treats them as spheres of a radius larger than its input radius. This representation is purely limited to the steps used for computing atom neighbor lists for reasons that are explained below. Once the neighbor list is created, the calculations of volume and surface are undertaken using a Gaussian density-based representation of the atoms.

To provide the exact schematic of our method and its workflow, we use an example molecule of five atoms. Without any

loss of generality, the grids will be portrayed in 2D and the atoms will be described as circles of radius equal to their vdW radius. This is illustrated in Figure 1. In the figure, the flow of steps is represented by a number on each of the panel, going from "1" through "6".

Step 1. A mesh, large enough to encompass all the atoms of the input molecule, is defined. A labeling system is used wherein each grid point is labeled by an integer. To initialize our grid, we assign "0" to each grid point.

Step 2. A separate $(N + 1) \times (N + 1)$ square matrix, depicting atom pairs that overlap in space is defined. We will refer to the matrix as the *atom-overlap matrix* or AOM. All the atoms in the molecule with indices 1, 2, ..., N are considered along with a dummy atom of index 0. An element of this matrix is defined as $AOM_{m,n} \in [True, False] \forall m, n \in [0, 1, 2, \dots, N]$ such that if atoms m and n overlap in space, $AOM_{m,n} = True$ otherwise *False*.

Step 3. The first atom (with index "1") in the list is placed onto the grid. As the grid-points in the vicinity of atom 1, contained in its local cubic box (shown as squares in the figure), are surveyed by Delphi, grid points that lie within a distance of kR_1 from the center of atom 1 (\vec{r}_1), that is, those that satisfy the distance criterion $|\vec{r}_{\text{grid}} - \vec{r}_1| \leq kR_1; k \in \mathbb{Z}^+$, are made to undergo a

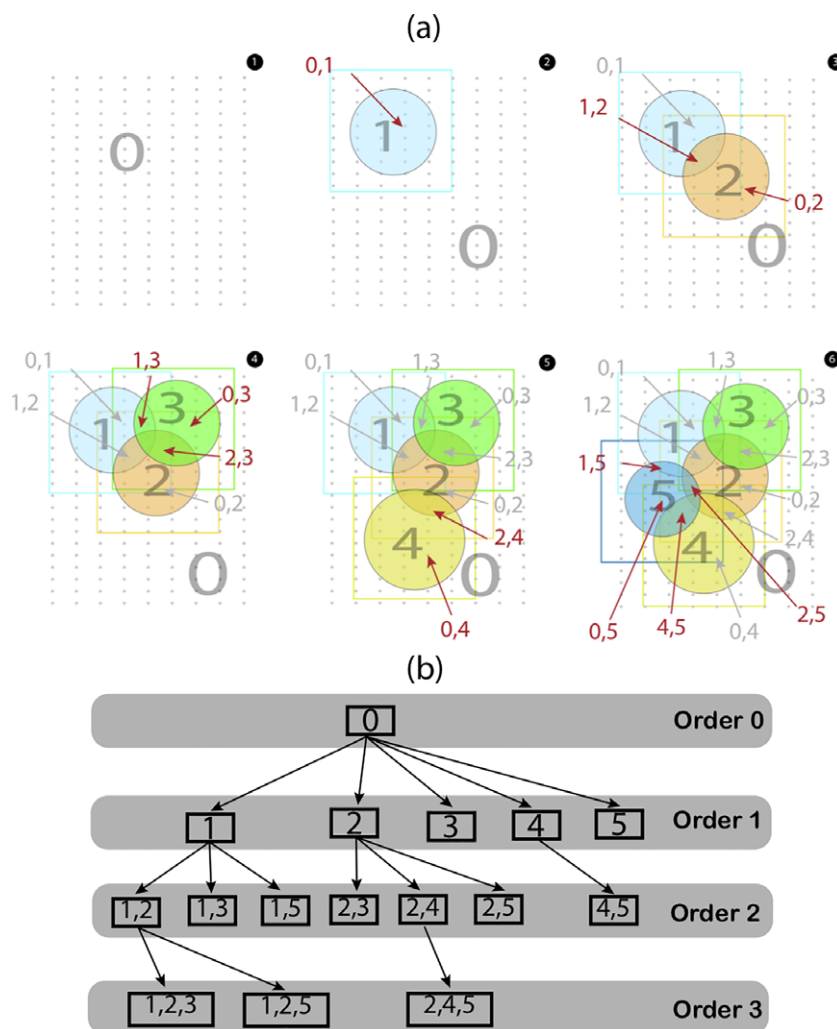


Figure 1. a) An illustration of the grid-based algorithm designed for identifying atom pairs that overlap in space. Each atom is shown as a colored circle surrounded by a square of the same color depicting the local box that is searched for grid-points in its vicinity. The systematic flow of the steps is indicated by the label on the top-right corner of each panel in the figure. Two atoms " i " and " j " that overlap update the atom overlap matrix (AOM) element $AOM_{i,j}$ to *True*. At each step, new indices of AOM that get updated to *True* are shown in red. The numeric labels placed at different regions are meant to indicate the integer label on the grid-points present in a region. b) A rooted tree constructed using the neighbor list of all the atoms in the molecule and an additional dummy atom with index "0". Each level or order is marked using gray horizontal bars. From top to bottom, levels of increasing orders are shown. [Color figure can be viewed at [wileyonlinelibrary.com](http://www.wileyonlinelibrary.com)]

Table 1. The atom overlap matrix or AOM (top panel) and the neighbor list of atoms inferred from it (bottom panel) for the five-atom example molecule obtained using the grid-based neighbor search algorithm. For clarity, only the upper triangular part of the symmetric matrix is shown.

Atom overlap matrix (AOM)						
	0	1	2	3	4	5
0	–	T	T	T	T	T
1		–	T	T	–	T
2			–	T	T	T
3				–	–	–
4					–	T
5						–

Neighbor list of the atoms	
Atom index	List of neighboring atoms
0 (Dummy Atom)	[1, 2, 3, 4, 5]
1	[2, 3, 5]
2	[3, 4, 5]
3	[]
4	[5]
5	[]

"True" is represented as "T" and "False" is represented as "–".

change in their integer label. "k" here is a factor that affects the volume of the box that is searched for grid points that fit the criteria. From the initial "0" label, they are assigned a label of "1" as they lie in the vicinity of atom 1. This change in label of the grids is accompanied by updating $AOM_{0,1} = \text{True}$. Essentially, the matrix element with row-index equal to the old label and column-index equal to the new label is updated to *True*.

Step 4. The second atom (with index 2) is placed onto the grid. Grid points that satisfy the above distance criterion with respect to atom 2, are surveyed and their labels are updated accordingly. Those with "0" are now labeled as "2", causing $AOM_{0,2} = \text{True}$ and those with "1" are now labeled as "2", causing $AOM_{1,2} = \text{True}$.

That $AOM_{1,2} = \text{True}$ exists implies that atoms 1 and 2 potentially overlap.

Step 5. The third atom (index 3) is placed onto the grid. At this point, grid points are labeled as either '0' or '1' or '2'. Grid points satisfying the distance criteria with respect to atom-3 result into updating $AOM_{0,3}$, $AOM_{1,3}$ and $AOM_{2,3}$ to *True*.

Step 6. Similarly, atom 4 and 5 are treated and the corresponding elements in the AOM are updated.

It must be noted here that as atoms are used in an increasing order of their index, the above procedure will only update the upper triangular block of the AOM. This does not result in losing any information because if atoms "m" and "n" overlap (where $m < n$) due to $AOM_{m,n} = \text{True}$, then it directly implies that $AOM_{n,m} = \text{True}$.

The final AOM is used to prepare the neighbor list. The following steps are performed.

Step 1. For each atom, an empty neighbor list (to store integers or atom-indices) is defined.

Step 2. An iterator navigates through the upper triangular part of the symmetric AOM and checks for all the elements $AOM_{m,n} \mid m \leq n$ which are *True*.

Step 3. For any $AOM_{m,n} = \text{True}$, index *n* is appended to the neighbor list of atom *m*.

For our example molecule with five atoms, Table 1 shows the AOM (in the upper-triangular form) and the list of neighbors for each atom. The outcome can be confirmed by the arrangement of atoms in Figure 1a. Also note that, atom with index "0" is a dummy atom and it automatically has all the "real" atoms of the molecule in its neighbors list. This helps in the construction of a rooted tree that is used for computing overlap volumes and SA.

A depth-first traversal algorithm for computing total volume and SA

The neighbor lists of the atoms are used to construct a rooted tree of overlaps, the hierarchy of which follows the order *n* of the overlaps. Each node of the tree holds the value of the overlap volume which is arithmetically added, according to eq. (9), to yield the total MV. For computational efficiency, overlap volume terms with values less than 0.001 \AA^3 are neglected. A volume cutoff of this kind is necessary as the Gaussian overlap volume of two distant atoms, albeit infinitesimally small, will never be zero. In parallel to volume calculations, the SA term for each node is also computed and the total molecular SA is obtained.

The basic premise of constructing a tree by a "depth-first" algorithm and using it for volume/SA computation is identical to the one used in reference 39. Each atom is assigned an integer index (starting from 1) and a dummy atom with index "0" is used to build a rooted tree with it being the designated root. Each subsequent level in the tree is assigned an order based on its distance from the root; the root is assigned an order 0 at the first step of the process. All the atoms are then defined as the children of the root, hence, forming the next level down the hierarchy with order 1. Each of these atoms then initiate a separate branch of the tree. The tree grows more levels by incorporating new nodes of the next order that contains the information of all the common neighbors of its ancestors. Eventually, a node of any order is designed to contain the information of all the neighbors common between itself and its ancestors. Computationally, a node of order "k" is represented by an ordered list of 'k' atom indices such that the atom with the kth index is a common neighbor to all the "k–1" atoms preceding it. Geometrically, that implies that all the *k*-atoms overlap in space. For example, if a node (1, 2, 3, 7) exists for an arbitrary molecule, it would imply that atom-7 is a common neighbor of atoms-1, 2, and 3. It would also mean that the four atoms overlap in space. A branch of the tree is terminated when a new common neighbor is not found or when the volume of that particular node is smaller than the cutoff value (0.001 \AA^3 , see above). For computational efficiency, we limited the order of nodes to 6. As the branch reaches a "dead-end", the next branch from the top of the tree is worked upon in the same recursive manner till all the branches growing out of the root have been covered. For our example case of the five-atom molecule, Figure 1b is an illustration of its depth-first tree.

It must be noted that, though the Gaussian-model projects a physically meaningful picture of a protein-solvent system, the mathematical formulation can harbor some unphysical issues.

Therefore, it is necessary that they are eliminated correctly. An example is negative SAs for deeply buried atoms surrounded by many neighboring atoms.^[39] For such atoms, it is likely that certain orders of the overlap volume, which have a negative contribution to its total SA, add up to be larger than its individual volume (e.g., order 2). To correct for this, we devised a physically appealing way of filtering the contribution of these atoms to the total SA. This filter uses a smooth sigmoid function of the form:

$$SA_{\text{filtered},i} = SA_i \left(\frac{1}{1 + e^{g(-SA_i + SA_{\text{cutoff},i})}} \right) \quad (13)$$

where '*i*' depicts an atom and SA_i is the surface area computed by the Gaussian model. '*g*' is a dimensionless constant with a value 5, assigned after optimization. $SA_{\text{cutoff},i}$ is a threshold value of the SA which decides if an atom contributes to the total SASA of the molecule. Only the atoms with values larger than the cutoff contribute. The cutoff is computed using a hard-sphere approximation and hence depends on the radius of a solvent-probe (R_{probe} ; 1.4 Å for water) and the radius of that atom (R_i). An atom is considered solvent accessible if it can allow at least one solvent molecule (in its hard sphere form) to share a tangential plane with it. The cutoff, therefore, acquires the following form:

$$SA_{\text{cutoff},i} = SA_i \left[\frac{1 - \cos(\theta)}{2} \right] \quad (14)$$

where the angle " θ " is the solid angle subtended by a cone of height $R_{\text{probe}} + R_i$ and base radius R_{probe} . It can be expressed as:

$$\theta = 2 \tan^{-1} \left(\frac{R_{\text{probe}}}{R_{\text{probe}} + R_i} \right) \quad (15)$$

Figure 2 provides a visual reference which exemplifies the case of a solvent of probe radius 1.4 Å and an atom of radius 2 Å.

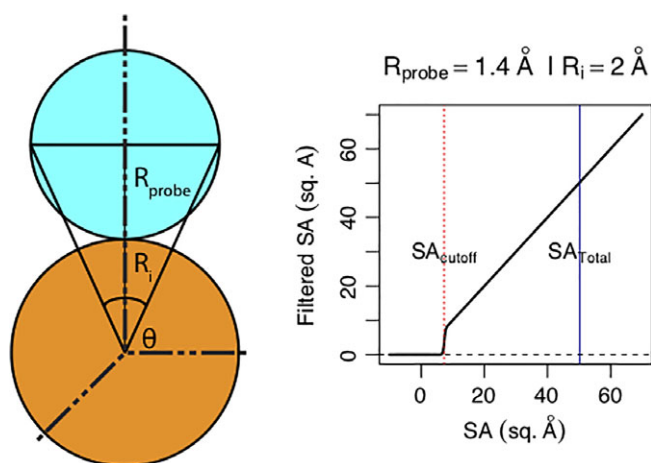


Figure 2. (Left) Illustration of the physical basis of the function used to compute cutoff atom-specific surface area and filter out the contribution of atoms with negative surface area terms. (Right) The output yielded by the filtering function. [Color figure can be viewed at wileyonlinelibrary.com]

Validation

We validated our grid-based approach at three different levels. First, the volumes and SAs for a library of 74 proteins of sizes ranging from 50 to 200 residues (used in a previous work^[63]) were calculated using algorithm as well as AGBNP^[37,39] and then were compared to determine the numerical differences. Second, the effect of grid-resolution on the output of volume and SA was examined. Third, the accuracy in identifying "correct" neighbors using the grid-based approach was evaluated by comparing the neighbors identified using a standard $O(N^2)$ analytical approach (see eq. (12)).

Figures 3a and 3b show the comparison of the volumes and SA of 74 proteins computed using our implementation of the Gaussian model and that of AGBNP. The quality can be adjudicated by the slope and intercept of a linear regression fit as well as the correlation (R^2) accompanying the figures. Slopes approximately equal to 1.00 (with relatively infinitesimal intercepts) and correlations equal to 1.00 indicate that our implementation is precise. In addition, we also acknowledge that the resolution of grids *a.k.a* "scale" in Delphi can have an effect on the volume/SA value by having an effect on the neighbor search process. Therefore, different values of scale were also used and the resulting volume outputs were compared with AGBNP. The results are shown in Figure 3c in terms of the root mean square relative difference (RMSRD; see Appendix A) incurred as a function of the grid resolution, which indicates that the differences are small, that is, ~0.40% at a low resolution of 1 grid/Å and ~0.15% at 2 grids/Å and become infinitesimal (<0.1%) at 3 and 4 grids/Å. But as increased resolutions mean nonlinear increase in computational times (cubic power), one should consider a balance between accuracy and computational time.

For the second level of validation, we examined the effect of differently positioning the solute inside the box without changing the position of the grids. This was important because in the initial phase of a Delphi run, the coordinates of a 3D structure (from PDB for instance) are projected onto these grids using a distance-dependent interpolation technique. For this test, we chose Barstar (PDB ID: 1X1X, chain D) and changed its position continually along an arbitrarily chosen direction (without loss of generality), by offsetting its coordinates from the center of the box in small incremental steps and computing the volume and SA using the Gaussian model. Figures 3d and 3e show the outcomes as a function of the offset distance for different values of scale. A tendency to vary periodically with regards to the offset is seen in these plots. The periodicity also varies with grid resolution, in that, the period is inversely proportional to the number of grids/Å. This is because with different offsets, the projection of the atom coordinates on the grids changes and this affects the neighbors identified in the process and congruent grid placements occur in multiples of the grid resolution. Eventually, that results into variations in the volume and SA outputs. But these variations are minor in comparison to the average values (<0.05%). This leads us to conclude that the grid-based approach is appreciably precise and is only minutely sensitive to the arrangement of the grid points in the box.

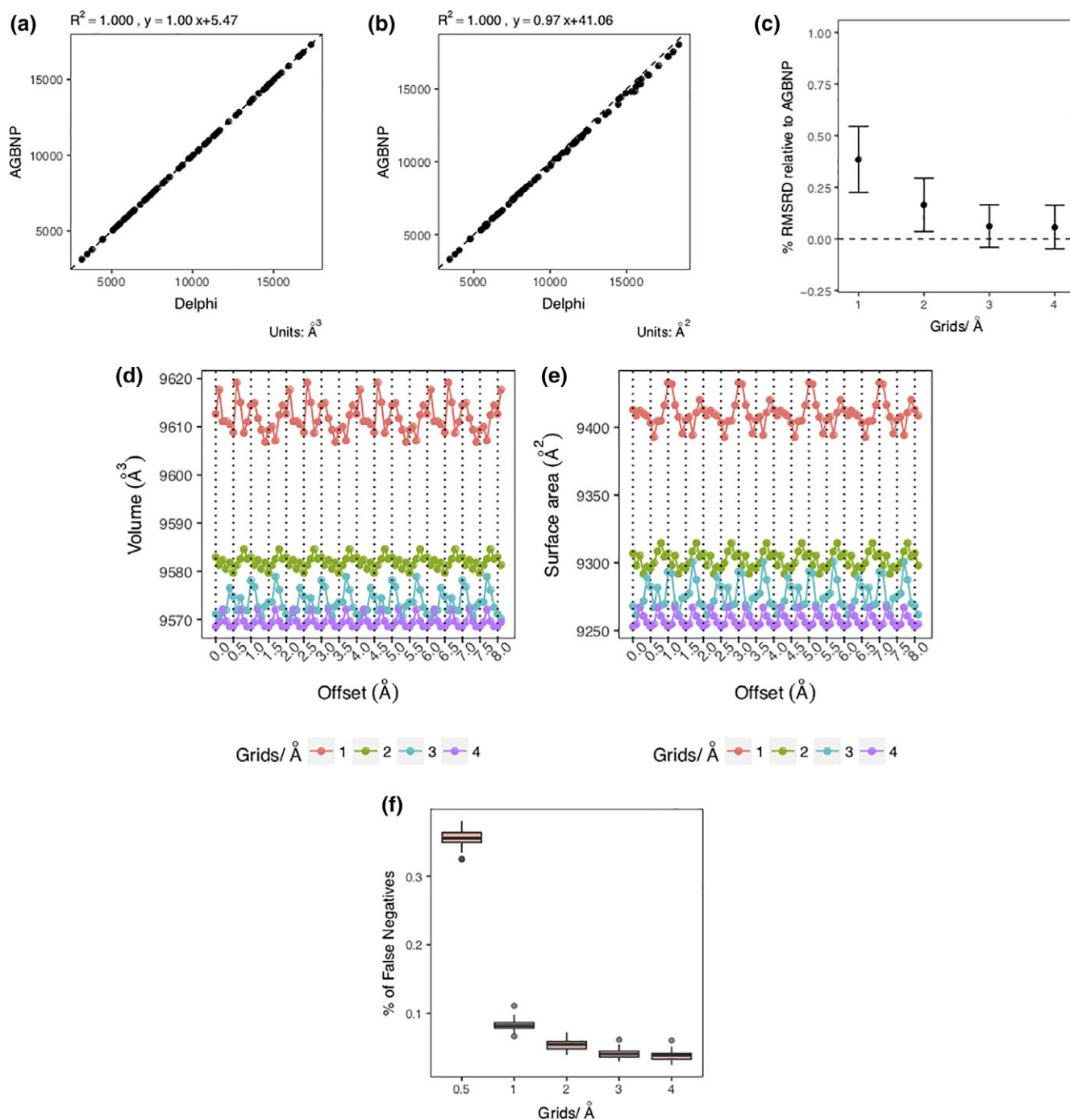


Figure 3. Validation of the grid-based algorithm for identifying atom pairs that overlap in space. Comparison of (a) the molecular volumes and (b) the molecular surface areas of 74 proteins obtained using the grid-based algorithm in conjunction with the Gaussian-model and obtained using AGBNP. (c) Percent relative difference (RMSRD) of the molecular volumes of the 74 proteins with respect to the values output by AGBNP as a function of the scale or grid-resolution. (d) Volume and (e) surface area of Barstar (PDB: 1X1X, chain D) plotted as a function of the offset in its position from the center of the grid box. (f) Percentage of falsely missed atom pairs overlapping in space (*False Negatives*) by the grid-based algorithm plotted as a function of the grid-resolution (grids/Å). [Color figure can be viewed at wileyonlinelibrary.com]

At the third level of validation, we evaluated our method's accuracy in determining the "correct" neighbors. In a standard approach, neighbor list for the atoms can be computed using a distance-based criterion where two atoms with coordinates separated by a distance lower than the sum of their radii are considered as neighbors (eq. (12)). But in our grid-based approach, two atoms are considered as neighbors if their box

of grids surrounding the respective spherical volumes shares common grid-points (see Algorithm). Therefore, we compared the neighbor list yielded by the grid-based approach, at different grid resolutions, with that obtained by using the standard $O(N^2)$ approach. This test was expected to report neighbors that are common to both the approaches (*True Positives*) or are neighbors based on one approach and not the other (*False*

Negatives or False Positives). Our grid-based approach would ideally “pass” the test if it can identify at the very least all the neighbors that the standard approach would. Any additional neighbors detected (*False Positives*) would later be filtered out based on the volume of their shared region (i.e., $<0.001 \text{ \AA}^3$). However, if a vast percentage of neighbors is only found by the standard approach and not by the grid-based approach (*False Negatives*), it would question the method’s credibility. Our focus is to detect the percentage of such cases. Figure 3f shows the outcomes. As a function of the grid resolution, the percentage of *False Negative* cases is plotted. Each boxplot depicts the range of percentage of *False Negatives* found across a library of 74 proteins and the solid black line close to the center of these boxplots is the median value of the distribution (see Appendix B). There are two major observations: 1) The percentage of the *False Negative* cases are infinitesimally small ($<0.4\%$) if not exactly 0.0 at a very coarse resolution of 0.5 grids/ \AA . 2) With finer resolutions, the percentage drops to $\sim 0.05\%$. This means that the grid-based approach could likely miss 1 in every 2000 neighbors identified using the standard approach. This imparts an added confidence in the accuracy of our approach.

Performance

We also assessed the time efficiency and complexity of the grid-based approach. Theoretically, it is an $O(8NR^3G^3)$ complex algorithm, where “ N ” is the number of atoms and “ R ” is the average atomic radius and “ G ” is the number of grids/ \AA . This is because for each atom out of N , a local cubical volume around its center is surveyed for the grid points which are later used by Delphi for assigning dielectric values and distributing charges. This local cube is of length proportional to $2R$ (average atomic diameter), making its volume $8R^3$ and the total number of grid points to be scanned equal to $8R^3G^3$.

But the integration of the grid-based algorithm in parallel with other grid-based operations performed by Delphi makes it difficult to evaluate the exact time. Therefore, we measured the total time taken by the grid-based neighbor search algorithm and the volume/SA computation using our implementation of the Gaussian model and subtracted it from the time taken by Delphi when these calculations were turned off. This gives an estimate of the average time efficiency as a function of grid resolution and size of the solute.

Figure 4 plots the average time over 10 runs versus the number of atoms for different grid resolutions. It is clear that time taken for volume/SA computation along with the neighbor search part is typically $<3 \text{ s}$ for proteins with 1000–3000 atoms when 1 or 2 grids are placed per \AA . Increasing the number of grids/ \AA (or resolution) appears to drastically increase the time. The effect is prominent when the number of atoms is more than 1000. This is because with increased resolution, the number of neighbors identified by our approach is much larger than that by the standard distance-based approach. In other words, the percentage of *False Positives* increases (see Fig. S1 of the Supporting Information).

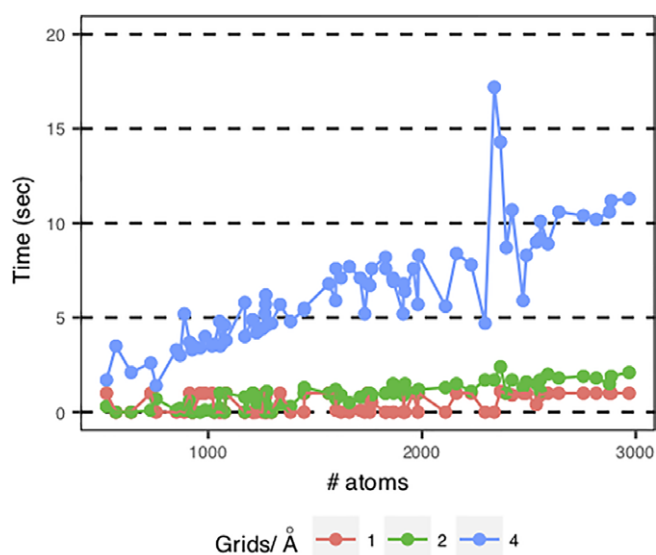


Figure 4. The average run time as a function of the number of atoms in the solute and grid resolution (grids/ \AA). Seventy-four proteins were used for the test and the average time was computed by averaging over 10 runs on each protein. As the standard deviations of the runtimes were infinitesimally small, error bars depicting them are deliberately not shown. [Color figure can be viewed at wileyonlinelibrary.com]

Results and Discussion

The contents of this section review basic aspects of the Gaussian model and in addition also points out other aspects that have not been addressed meticulously in the literature before. The idea is to emphasize on the numerical accuracy, physical appeal and an inevitable limitation of the model.

Volume and SA computed using the Gaussian model

The Gaussian model, as proposed by Grant and Pickup in their seminal work, delivers the vdWV and vdWSA of molecules.^[49] Using our implementation with the grid-based neighbor search algorithm and the library of 74 proteins, we found that the volumes delivered by it differ from the hard-sphere vdWV by $\sim 7\text{--}8\%$ (the latter was computed using the package ProteinVolume^[55]). We also used another package called 3V^[64] for a thorough benchmarking and found that the difference, in this case, was smaller (difference $< 1\%$). In terms of the SA, we found that the output from the Gaussian model differs by $\sim 6\text{--}6.5\%$ from the vdWSA computed using the hard-sphere model by FREESASA.^[65] Once again for a thorough benchmark, we also used NACCESS^[66] and the difference was found to be $\sim 4.5\%$. The exact values of these differences, expressed using RMSRD and the outcomes of a linear regression fit to the model comparisons are presented in Table 2. For all the Gaussian model-based calculations, a “ κ ” value of 2.227 was used^[37,39] (see eq. (2)) and a resolution of 2 grids/ \AA was used for grid-based neighbor search.

There are two major inferences we draw from these comparisons. First, our implementation of the Gaussian model precisely delivers the molecular vdWV and vdWSA indicating that the implementation correctly reproduces the expected behavior of the Gaussian model. Second, volumes/SA computed using the

Table 2. Comparison between van der Waals volumes and surface area of proteins and surface area of individual atoms obtained using the Gaussian model and the hard-sphere model. The comparison is quantified by the slope, intercept of the linear regression fit, correlation (R^2) and the root mean square relative difference (RMSRD).

	RMSRD (%)	Slope	Intercept	Correlation (R^2)
van der Waals volume of proteins				
ProteinVolume	7.70	1.08	7.40 Å ³	0.999
3 V	0.24	1.00	7.20 Å ³	0.999
van der Waals surface areas of proteins				
FREESASA	5.9	0.94	32.09 Å ²	0.999
NACCESS	4.3	0.95	64.51 Å ²	0.999
van der Waals surface area of individual atoms				
FREESASA	15.9	0.89	0.70 Å ²	0.953

hard-sphere model do not offer a strict reference for benchmark and validation. This is evident from comparing the results computed by different software. Indeed, different packages implementing the hard-sphere model yield different values (Table 2).

We further extended the comparison between the two models at the level of SAs of individual atoms. Using the two models, SA of individual atoms across the 74 proteins were computed and compared. The result, also shown in Table 2, clearly indicates that the Gaussian model can deliver precise SAs of individual atoms with a difference of only 15.9% with respect to the values computed using the hard-sphere model. This good quality of the agreement is also evident from the slope and intercept of the linear regression and a correlation of 0.953. This ability to deliver proper SAs of individual atoms provides the Gaussian model with an added advantage. Several packages like AGBNP^[37,39] and ACE,^[67] that run MD using the Gaussian model, make use of this ability to correctly compute the energy and forces on individual atoms alongside the continuous and differentiable analytical expressions for this terms. In addition to this, atom-specific surface-tension coefficients used in conjunction with individual atomic SAs has been shown to deliver nonpolar part of the free energy in good agreement with that from explicit solvent simulations.^[42]

Physical appeal

In addition to numerical precision, one of the key features of a smooth Gaussian-based model is that the transition area between solute and the solvent phases does not have to be sharp. To demonstrate this, we present a profile of the change in the vdWV/SA of a protein complex as a function of the distance between the monomers as they are separated in space. A test of the same nature was performed by Grant and Pickup in the process of parametrical optimization of the model.^[49] For our study, we separated chains A and D of the Barnase–Barstar complex (PDB ID: 1X1X) in steps of 0.1 Å starting from the bound state to 15 Å away. At 15 Å separation, the monomers are practically free (completely unbound).

The profiles are shown in Figure 5. Clearly, the change in vdWV/SA obtained using the Gaussian model (Figs. 5a and 5c) has an overall smooth trend (if one momentarily overlooks the periodic effects arising from use of the grid-based approach). In the completely unbound state, the volume and SA of the dimer

is simply equal to the sum of these quantities for the individual monomers. On the contrary, the profile obtained using the hard-sphere model features some prominent bumps, discontinuities, and noticeable regions of transitions. Between 1–2 Å of separation, the hard-sphere model yields an increase in the total volume of the complex and then it drops at around 2–3 Å (Fig. 5b). Following this drop, it again increases monotonically till the total volume saturates at a value that equals the sum of the volumes of the monomers. In terms of the SA, there is a drastic initial increase till the monomers are separated by approximately 1 Å after which there is a small discontinuity leading to a plateau in the profile (Fig. 5d). Once separated by approximately 3 Å, the profile acquires a monotonically increasing trend till it saturates to a value that equals the sum of the SAs of individual monomers (occurs at ~8 Å separation).

Overall, the major difference in the profiles from the Gaussian and the hard-sphere model occur at small separations. This can be attributed to the method used by these approaches to treat the intersection or overlap volumes. To elaborate, we plot the number of contacts between the two monomers (chains A and D) as a function of the distance of separation in Figure 5e. A *contact* here is defined as a pair comprised of an atom from Barnase and an atom from Barstar whose centers are separated by no more than 4 Å. As the monomers move farther apart, the number of contacts drops drastically at small separation distances and becomes zero at distances greater than 9 Å. The region in between exhibits several abrupt transitions (2–9 Å). The drastic and discontinuous drop in the region from 0–2 Å explains the abrupt changes in the collective volumes of overlapping atoms at the interface. This is the cause for the bumps in the profile obtained using the hard-sphere model at that region. This is also the region that emphasizes on the ability of the Gaussian model to deliver smoothly changing volumes. In case of the Gaussian model, the volume of the overlapping regions (regardless of the order) between the atoms at the interface has a continuous expression (see eq. (7)) which eventually renders a smooth change of the two quantities.

Gaussian model to compute solvent excluded volumes

The solvent excluded volume (SEV) and the corresponding surface area (SESA) are considered more faithful representations of the geometry of the solute–solvent interface than the vdW counterparts. These representations appropriately characterize those voids present in the solute structure, which are too small to fit a solvent molecule, as part of the solute phase. By virtue of this definition, SEVs are larger than the vdW volumes because the latter is a part of it. With our library of 74 proteins, SEVs were found to be 25–50% larger than their vdW volumes (average difference was ~38%).

In an attempt to enable the Gaussian model to deliver SEVs, Gallicchio et al.^[39] incorporated a modification in the Gaussian model. The modification involves augmenting the radius of all the solute atoms by an offset term R_{offset} so as to account for the volume of crevices in the total volume. To enhance the physical appeal of this modification, an additional correction/modification was later incorporated.^[37] The central idea of the

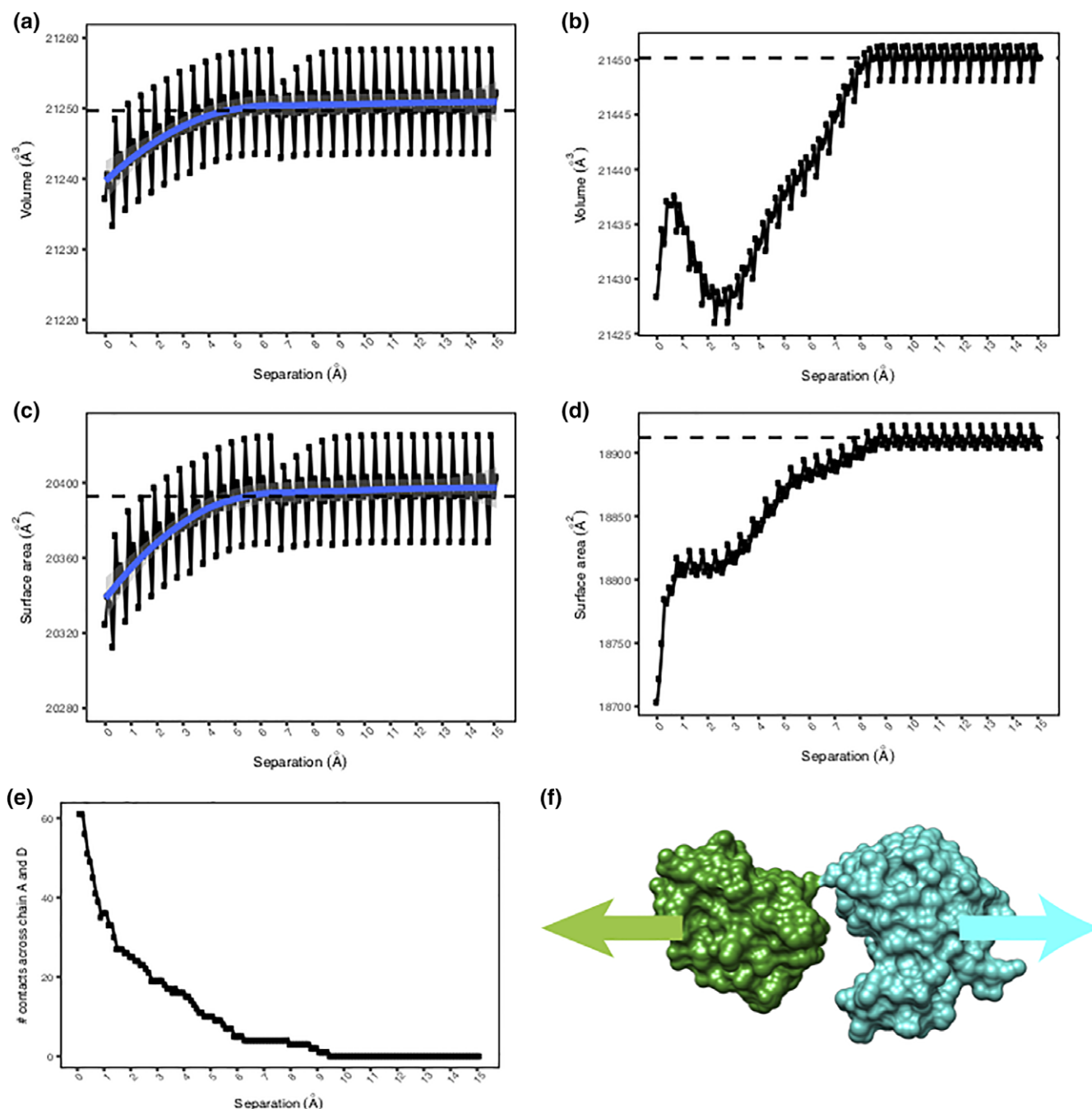


Figure 5. Profile of the change in the van der Waals (vdW) volume and surface of the Barnase–Barstar complex computed using two different models as a function of the distance of separation of the monomers. Profile of the change in vdW volume obtained (a) using the Gaussian model and (b) using the hard-sphere model. Profile of the change in vdW surface area obtained (c) using the Gaussian model trend and (d) using the hard-sphere model. The solid blue lines in (a) and (c) depict a nonlinear fit to the profiles obtained using the Gaussian model in order to emphasize the overall smoothness of the trend. The vdW volume and surface area using the hard-sphere models were computed using $3 V_i^{[64]}$ with a probe of radius 0.0 Å. (e) Change in the number of contacts, that is, atom pairs from either monomer found to be within 4 Å distance, as a function of the distance of separation of the monomers. (f) A cartoon representation of the setup in which the monomers of the Barnase–Barstar complex were separated for obtaining the above profiles of volume and surface area changes. [Color figure can be viewed at wileyonlinelibrary.com]

modification was motivated by the fact that as R_{offset} increases the atomic radii in order to account for the interstitial crevices in the structure, it also causes the solvent-exposed atoms to expand further into the solvent region. Therefore, the excess volume of the solvent exposed atoms can be discarded by computing the volume of only the solvent-exposed region of the atoms and subtracting it from their volume obtained with

modified atomic radii (V_i^{offset}). This is illustrated in Figure 6a and eq. (16) expresses this correction term.

$$V_i = V_i^{\text{offset}} - V_i^{\text{solvent-exposed region}}$$
$$V_i = \frac{SA_i^{\text{offset}}(R_i + R_{\text{offset}})}{3} \left(1 - \left(\frac{R_i}{R_i + R_{\text{offset}}} \right)^3 \right) \quad (16)$$

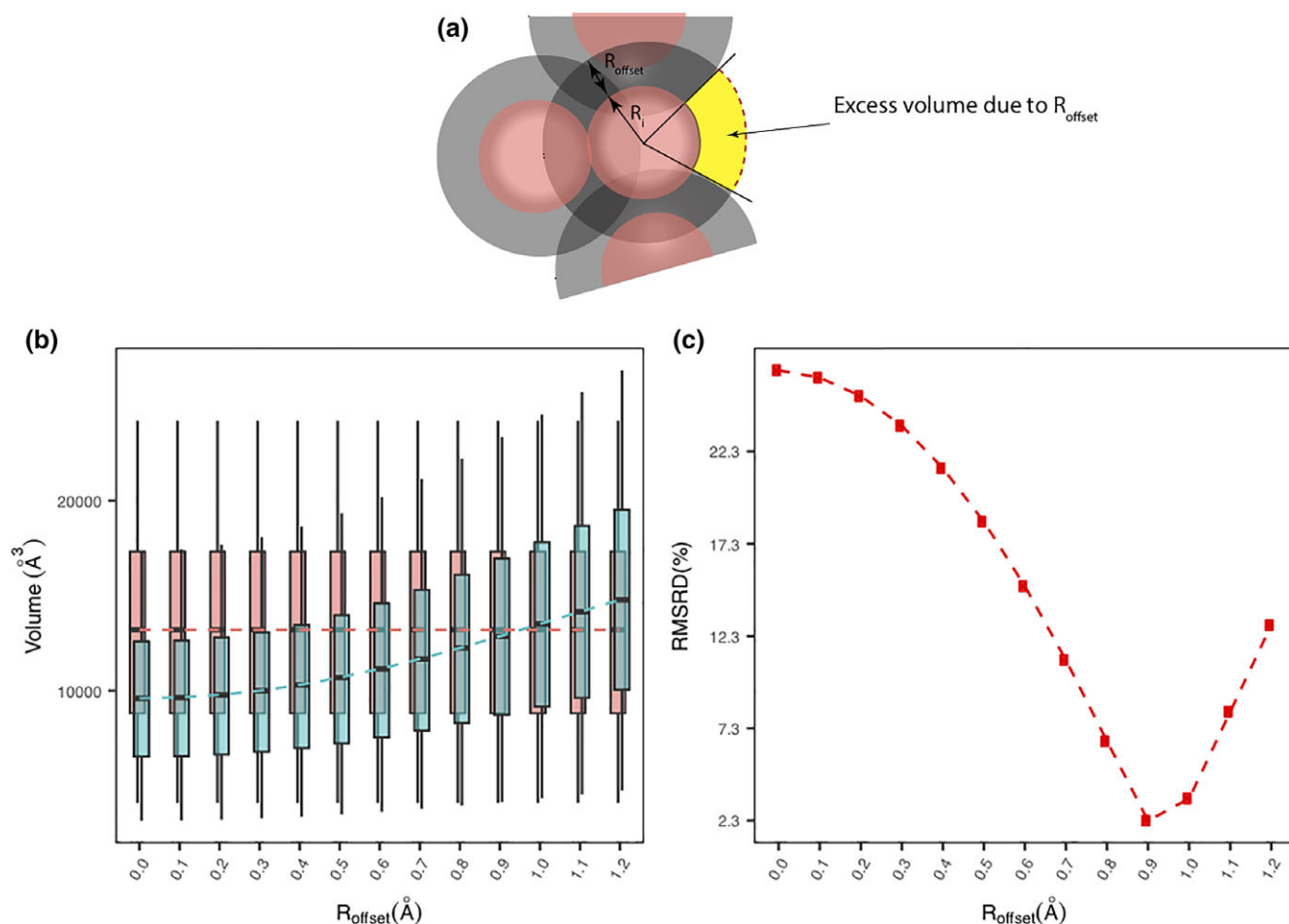


Figure 6. Optimization of R_{offset} input to the modified R_{offset} -based Gaussian model with respect to the SEV obtained using a hard sphere model. Distributions and relative percent deviations (RMSRD) are computed for the protonated and minimized crystal structures of 74 proteins. a) Schematic showing the basis of the modified R_{offset} -based Gaussian model in which the excess volume of a solvent exposed atom (shown in yellow), obtained by augmenting its van der Waals radius by some R_{offset} , is subtracted out when the correction is applied. b) Distribution of volume output by the modified R_{offset} -based Gaussian model (blue) computed with various R_{offset} values ranging from 0.0 through 1.2 \AA , compared with the distribution of the hard-sphere solvent excluded volumes (pink) for the same set of proteins. Each distribution is represented by a boxplot (see Appendix B). The dashed lines connecting the medians of the boxes highlight the overall trend. (b) %RMSRD of the volume from the Gaussian model with respect to the SEV as a function of R_{offset} . [Color figure can be viewed at wileyonlinelibrary.com]

The expression in eq. (17) ensures that the correction is only applied to the solvent exposed atoms with $SA_i^{\text{offset}} \neq 0$ computed using augmented atomic radii.

We added this modification in our implementation of the Gaussian model and evaluated a series of values of R_{offset} to find the value that yields the best agreement with the SEV computed using hard-sphere model with a solvent probe of radius 1.4 \AA . R_{offset} was systematically varied from 0.1–1.2 \AA and we found that 0.9 \AA gives the best agreement with a RMSRD of 2.3% (Figs. 6b and 6c). The goodness of the agreement is also confirmed by the quality of the linear regression which has a slope of 0.95, y-intercept of 326.22, and correlation of 1.00. In Supporting Information Table S1, we list the slope and intercept of the linear regression, correlation, and the RMSRD for all the R_{offset} values in this range.

Although this empirical approach delivers a good agreement, the analysis so far as only emphasized on the numerical aspect. It was, therefore, important to examine if this modification is

realistic in nature and if it retains the physical appeal of the Gaussian model. Two different approaches were used to address this. We examined (1) if this modification truly accounts for the volume of the crevices in the structure and (2) if offers a physically meaningful description of the SEV.

Volume of interstitial regions

A simple formula was used to derive the volume of the interstitial regions of solutes. By subtracting out the volume obtained with no R_{offset} (original model) from the volume obtained with a nonzero R_{offset} , the interstitial volume or $V_{\text{interstitial}}$ were obtained (eq. (17)).

$$V_{\text{interstitial}}(R_{\text{offset}}) = \text{Volume}|_{R_{\text{offset}} \neq 0} - \text{Volume}|_{R_{\text{offset}} = 0} \quad (17)$$

By using the interstitial volumes obtained with the hard-sphere model using ProteinVolume^[55] as a reference, the

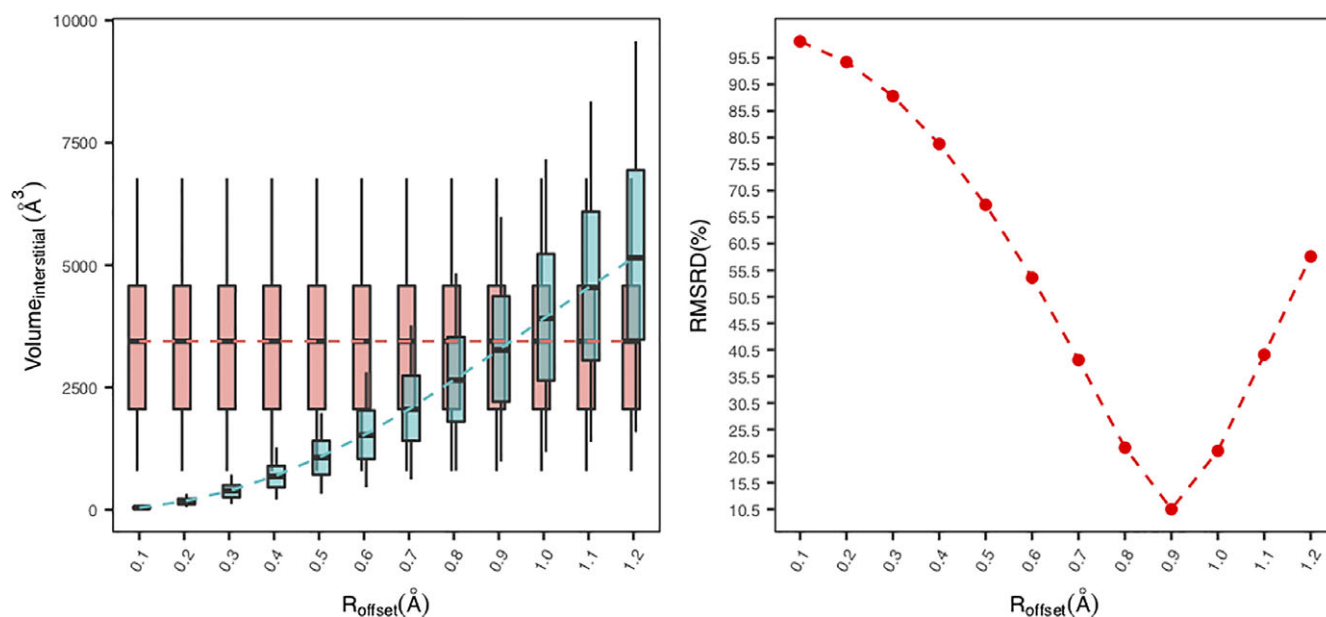


Figure 7. Comparison of the volume of the interstitial regions in the structure ($\text{Volume}_{\text{interstitial}}$) obtained using the modified R_{offset} -based Gaussian model (see eq. (17)) and the hard-sphere model. Distributions and relative percent deviations (RMSRD) computed for the protonated and minimized crystal structures of 74 proteins. (a) Distribution of $\text{Volume}_{\text{interstitial}}$ computed using the modified R_{offset} -based Gaussian model (blue) computed with various R_{offset} values ranging from 0.0 through 1.2 \AA , compared with the distribution of $\text{Volume}_{\text{interstitial}}$ computed using the hard-sphere model by ProteinVolume^[55] (pink). Each distribution is represented by a boxplot (see Appendix B). The dashed lines connecting the medians of the boxes highlight the overall trend. (b) %RMSRD of the $\text{Volume}_{\text{interstitial}}$ from the Gaussian model with respect to the $\text{Volume}_{\text{interstitial}}$ from hard-sphere model as a function of R_{offset} . [Color figure can be viewed at [wileyonlinelibrary.com](#)]

interstitial volumes computed using R_{offset} of 0.9 \AA were found to lie within 10.5%. In addition, a linear regression fit yielded a slope of ~ 0.81 and a correlation of ~ 0.99 . That the other values of R_{offset} did not perform as good as 0.9 \AA , as is evident from Figure 7, confirms that the numerical match obtained with this R_{offset} has a realistic foundation as well. In Supporting Information Table S2, we list the slope and intercept of the linear regression, correlation, and the RMSRD for all the R_{offset} values

that were used. The above calculations were done on the library of 74 proteins.

Physical appeal of the R_{offset} -based Gaussian model

In terms of physical appeal, we used the same case of Barnase–Barstar complex, separating the monomers from their bound state to a completely unbound state, to profile the change in

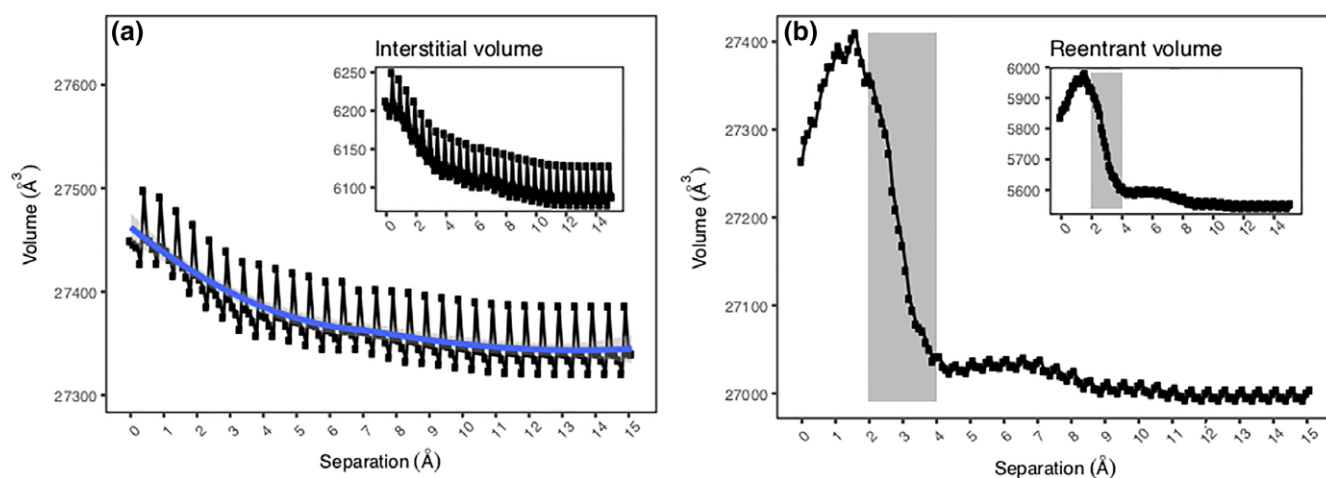


Figure 8. a) Change in the volume of the Barnase–Barstar complex output by the modified R_{offset} -based Gaussian model. The solid blue line depicts a smooth fit to emphasize the smooth trend. *Inset:* Difference of the volume output by the modified R_{offset} -based and the unmodified Gaussian model, which is supposed to depict the volume of solvent inaccessible crevices in the complex's structure, as a function of separation distance. b) Change in the solvent excluded volume (SEV) of the complex computed using 3 $V^{[64]}$ with a probe of radius 1.4 \AA as a function of the separation distance of the monomers. *Inset:* Volume of reentrant regions and solvent inaccessible crevices obtained by subtracting the van der Waals volume of the dimer from its SEV. The shaded region (gray) emphasizes the length scale of separation that is comparable to the diameter of the solvent probe (2.8 \AA). [Color figure can be viewed at [wileyonlinelibrary.com](#)]

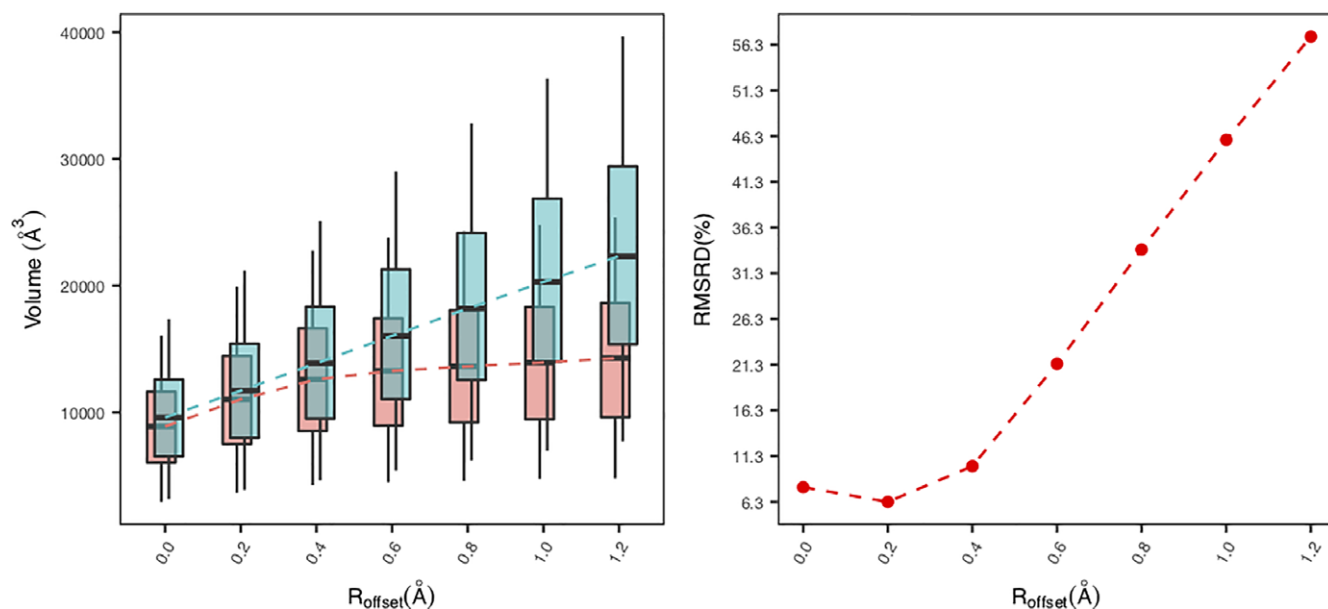


Figure 9. Comparison of the van der Waals volume from the R_{offset} -based Gaussian model (without correction of the excess solvent-exposed volume) and hard-sphere models when augmented radii for atoms are used. Distributions and percent deviations (RMSRD) are computed for the protonated and minimized crystal structures of 74 proteins. a) Distribution of volume output by the R_{offset} -based Gaussian model (blue) computed with various R_{offset} values ranging from 0.0 through 1.2 Å, compared with the distribution of hard-sphere volumes (pink) computed using the same set of augmented radii. Each distribution is represented by a boxplot (see Appendix B). The dashed lines connecting the medians of the boxes highlight the overall trend. b) %RMSRD of the volume obtained using the R_{offset} -based Gaussian model with respect to the volume output by the hard-sphere model as a function of R_{offset} . [Color figure can be viewed at wileyonlinelibrary.com]

volume. Except in this particular study, we replaced the vdW volume with the SEV from hard-sphere model and the volume from the modified R_{offset} -based Gaussian model. For the latter, we used R_{offset} of 0.9 Å. The profiles are shown in Figure 8. Clear differences are visible in the trends obtained from the two models. That the profile of the volume from the modified R_{offset} -based Gaussian model has a better physical foundation than the hard-sphere model is justified in the following two paragraphs.

The volume from the modified R_{offset} -based Gaussian model features a smooth monotonic decrease from an initial value to the value that equals the sum of the volumes of the individual monomers (Fig. 8a). The inset in the plot shows the volume derived from eq. (17) and shows that the excess volume computed by the R_{offset} -based Gaussian model (after the correction of the excess solvent exposed volume) monotonically and smoothly decreases as the separation increases. This smooth decrease can be better understood in terms of the change of average dielectric properties of the region between the monomers. As the monomers move apart, they gradually allow solvent molecules to occupy this region. But as the solvent molecules begin to enter the space between the interfaces, the interfacial residues from either monomer are expected to favorably interact with it to compensate for the loss of favorable interactions in the bound state. Consequently, the solvent molecules are not as mobile as their counterparts in the bulk and, therefore, tend to have a lower dielectric response, as has also been observed experimentally.^[68] This is the foundation reflected in the Gaussian-based smooth dielectric model proposed by us.^[47]

For the case of hard-sphere model, on the other hand, the volume increases from the initial value in the bound state till the separation is approximately 2 Å. Subsequently, there is a drastic drop in the volume in the region from 2–4 Å (shaded region in Fig. 8b). The size of this region is typical of the solvent probe's diameter and drop occurs because the concave reentrant surfaces, previously bounding the solvent inaccessible crevices at the interface between the monomers, disappear at this degree of separation. The inset plot in Figure 8b shows this loss of the solvent inaccessible volume bound by the reentrant surfaces. This volume is simply derived by subtracting the SEV of the dimer system from its vdW volume. The sudden loss of reentrant volume at the interfacial region implies that solvent molecules can enter this region and retain the dielectric response seen in the bulk phase. Although, this model of dielectric distribution is conventional in PB modeling of solvated dimer systems, it fails to capture the possibility of interaction of the newly exposed interfacial residues with the solvent.

Limitations to the applicability of the Gaussian model with large R_{offset}

There is a practical risk associated with using radius offsets comparable in magnitude to the radius of atoms (e.g., R_{offset} of 0.9 Å). The Gaussian model of Grant and Pickup was designed and optimized to deliver vdW volume and SA but only in the limit of weakly overlapping atoms.^[69,70] Thus, augmenting the atomic radius also increases the degree of overlap of atoms and this brings the Gaussian model very

close or likely beyond its limit of applicability. With large overlaps and by virtue of the Gaussian product theorem (eq. (4)), the volume of the overlapping regions is overestimated with respect to what a hard-sphere model would deliver with the same set of augmented radii. Mathematically, as the overlapping region of any two atoms grows in volume, the volume of the atom pair grows proportionally to the product of the volumes of the individual atoms (V^2 , where V is the volume of one atom). Geometrically, however, if two atoms of similar volumes overlap significantly in space, the volume of the atom pair is proportional to the volume of the larger of the two atoms (or V). This fundamental problem can lead to errors in volume and SA estimates.

To test the effect of offsets, we computed the vdW volume using the Gaussian model and the hard-sphere model when both the models were provided with augmented atomic radii. This deliberately increased the degree of overlap of atoms due to their increased radii. By systematically varying R_{offset} from 0.0–1.2 Å, their distribution was compared and the relative differences were measured (Fig. 9). The overall trend indicates that as the value of R_{offset} is increased, the Gaussian and hard-sphere volumes start to deviate appreciably. Volumes obtained from the Gaussian model increase exponentially while volumes obtained using the hard-sphere model saturate after a certain point. With no offset, the volumes from the two models deviate only by ~7% (the difference in the vdW volumes) but this increases to ~41% when the radii are augmented by an offset of 1.0 Å and to ~55% when augmented by an offset of 1.2 Å. This exponential deviation reflects the overestimation of the overlap volumes that is geometrically incorrect.

This aspect of the Gaussian model will pose methodological issues if it used to compute the SASA of solutes using the definition used by the hard-sphere model. By that definition, SASA is essentially the vdWSA obtained when the radius of each atom is augmented by the radius of the solvent probe (typically 1.4 Å). But if an R_{offset} value of 1.4 Å is used in order to obtain SASA using the Gaussian model, it will be asked to operate beyond its range of applicability. Although this idea was titillated by Grant and Pickup in their original work,^[49] Weiser et al.^[69] emphasized on the issue with the approach. Weiser et al.^[69] also discussed other parametrical modifications to obtain SASA but carefully described their limitations too.

Conclusion

This work presents a novel grid-based algorithm of identifying overlapping pairs of atoms in conjunction with the analytical approach of a Gaussian-based model^[49] for computing MVs and MSAs. The primary motivation for this design is to integrate into Delphi,^[11] a popular PBE solver, a new feature for determining nonpolar parts of the free energy. This grid-based algorithm makes a simultaneous use of a cubic 3D grid-map constructed for Delphi's finite-difference based operations and by doing so, it incurs very little to no time in identifying the pairs of atoms that overlap in space. The validation of the grid-based algorithm


in terms of the final volume/SA output, accuracy in identifying overlapping atom pairs and time efficacy shows that the method is robust and credible for an integrated use with future versions of Delphi for MM/PBSA analyses. The integration of the Gaussian-based model of volume/SA with the Gaussian-based model of dielectric distribution of Delphi^[47] also promotes a description of solvated biomolecular systems which removes sharp surfaces separating the solute and the solvent phases and depicts a smoother transition instead. This work brings us one step closer to having an integrated platform for MM/PBSA calculations using a physically appealing, surface-free approach to evaluate the thermodynamics of solvation, binding and folding/unfolding of proteins in the framework of implicit solvent models.

Acknowledgments

Clemson University is also acknowledged for the generous allotment of computer time on its Palmetto cluster. The work of A.C. and E.A. was supported by a grant from NIH, grant number R01GM093937.

Keywords: molecular · volume · surface area · grid-based · Gaussian-based model

How to cite this article: A. Chakravorty, E. Gallicchio, E. Alexov. *J. Comput. Chem.* **2019**, *40*, 1290–1304. DOI: 10.1002/jcc.25786

 Additional Supporting Information may be found in the online version of this article.

Appendix A.

Root Mean Square Relative Difference (RMSRD)

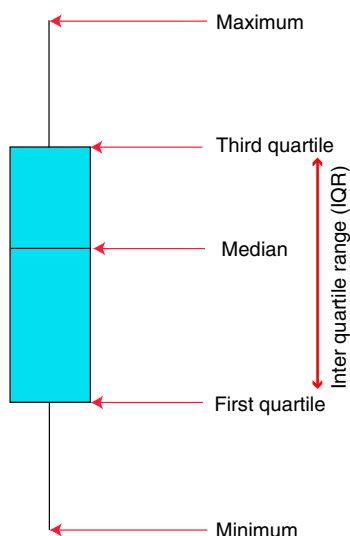
The expression for the RMSRD between two sets of data, X and Y , relative to one of them (say X) with the same strength, N , is given by the following expression:

$$\text{RMSRD} = 100 * \sqrt{\frac{\sum_{i=1}^N \left(\frac{X_i - Y_i}{X_i} \right)^2}{N}}$$

Appendix B.

Interpreting Boxplots

Boxplots are a useful way of representing a distribution. By depicting the different quantiles for the underlying data, they provide a better sense of the distribution. Presenting the mean and the variance of a data assumes that the data is normally distributed, which, however, is not always the case. Boxplots do not assume the category of the distribution of the data and can provide more information than just the mean and the variance. The figure below provides a guide to interpreting boxplots.



- [1] R. M. Levy, E. Gallicchio, *Annu. Rev. Phys. Chem.* **1998**, 49, 531.
- [2] A. L. Fink, *Fold. Design* **1998**, 3, R9.
- [3] K. A. Dill, *Biochemistry* **2002**, 29, 7133.
- [4] M. Petukh, L. Dai, E. Alexov, *Int. J. Mol. Sci.* **2016**, 17, 547.
- [5] Y. Peng, L. Sun, Z. Jia, L. Li, E. Alexov, *Bioinformatics* **2017**, 34, 779.
- [6] I. Getov, M. Petukh, E. Alexov, *Int. J. Mol. Sci.* **2016**, 17, 512.
- [7] K. J. Frye, C. A. Royer, *Protein Sci.* **1998**, 7, 2217.
- [8] J. Roche, J. A. Caro, D. R. Norberto, P. Barthe, C. Roumestand, J. L. Schlessman, A. E. Garcia, B. Garcia-Moreno, E. C. A. Royer, *Proc. Natl. Acad. Sci.* **2012**, 109, 6945.
- [9] J. Srinivasan, T. E. Cheatham, P. Cieplak, P. A. Kollman, D. A. Case, *J. Am. Chem. Soc.* **1998**, 120, 9401.
- [10] P. A. Kollman, I. Massova, C. Reyes, B. Kuhn, S. Huo, L. Chong, M. Lee, T. Lee, Y. Duan, W. Wang, O. Donini, P. Cieplak, J. Srinivasan, D. A. Case, T. E. Cheatham, *Acc. Chem. Res.* **2000**, 33, 889.
- [11] L. Li, C. Li, S. Sarkar, J. Zhang, S. Witham, Z. Zhang, L. Wang, N. Smith, M. Petukh, E. Alexov, *BMC Biophys.* **2012**, 5, 9.
- [12] D. A. Case, T. E. Cheatham, T. Darden, H. Gohlke, R. Luo, K. M. Merz, A. Onufriev, C. Simmerling, B. Wang, R. J. Woods, *J. Comput. Chem.* **2005**, 26, 1668.
- [13] D. Chen, Z. Chen, C. Chen, W. Geng, G.-W. Wei, *J. Comput. Chem.* **2011**, 32, 756.
- [14] N. A. Baker, D. Sept, S. Joseph, M. J. Holst, J. A. McCammon, *Proc. Natl. Acad. Sci.* **2001**, 98, 10037.
- [15] J. A. Grant, B. T. Pickup, A. Nicholls, *J. Comput. Chem.* **2001**, 22, 608.
- [16] C. M. Cortis, R. A. Friesner, *J. Comput. Chem.* **1997**, 18, 1591.
- [17] A. Onufriev, D. Bashford, D. A. Case, *Proteins Struct. Funct. Bioinf.* **2004**, 55, 383.
- [18] W. C. Still, A. Tempczyk, R. C. Hawley, T. Hendrickson, *J. Am. Chem. Soc.* **1990**, 112, 6127.
- [19] G. D. Hawkins, C. J. Cramer, D. G. Truhlar, *Chem. Phys. Lett.* **1995**, 246, 122.
- [20] R. B. Hermann, *J. Phys. Chem.* **1972**, 76, 2754.
- [21] D. J. Tannor, B. Marten, R. Murphy, R. A. Friesner, D. Sitkoff, A. Nicholls, B. Honig, M. Ringnalda, W. A. Goddard, *J. Am. Chem. Soc.* **1994**, 116, 11875.
- [22] T. Simonson, A. T. Bruenger, *J. Phys. Chem.* **1994**, 98, 4683.
- [23] C. Choithia, *Nature* **1974**, 248, 338.
- [24] J. Gelles, M. H. Klapper, *Biochim. Biophys. Acta* **1978**, 533, 465.
- [25] L. Chiche, L. M. Gregoret, F. E. Cohen, P. A. Kollman, *Proc. Natl. Acad. Sci. USA* **1990**, 87, 3240.
- [26] J. A. Reynolds, D. B. Gilbert, C. Tanford, *Proc. Natl. Acad. Sci. USA* **1974**, 71, 2925.
- [27] K. Lum, D. Chandler, J. D. Weeks, *J. Phys. Chem. B* **1999**, 103, 4570.
- [28] E. Gallicchio, M. M. Kubo, R. M. Levy, *J. Phys. Chem. B* **2000**, 104, 6271.
- [29] D. L. Mobley, C. I. Bayly, M. D. Cooper, M. R. Shirts, K. A. Dill, *J. Chem. Theory Comput.* **2009**, 5, 350.
- [30] Y. K. Kang, G. Nemethy, H. A. Scheraga, *J. Phys. Chem.* **1987**, 91, 4105.
- [31] J. A. Wagoner, N. A. Baker, *Proc. Natl. Acad. Sci. USA* **2006**, 103, 8331.
- [32] C. Tan, Y.-H. Tan, R. Luo, *J. Phys. Chem. B* **2007**, 111, 12263.
- [33] G. Hummer, *J. Am. Chem. Soc.* **1999**, 121, 6299.
- [34] P. Ferrara, J. Apostolakis, A. Caffisch, *Proteins Struct. Funct. Genet.* **2002**, 46, 24.
- [35] R. M. Levy, L. Y. Zhang, E. Gallicchio, A. K. Felts, *J. Am. Chem. Soc.* **2003**, 125, 9523.
- [36] J. Chen, C. L. Brooks III, *Phys. Chem. Chem. Phys.* **2008**, 10, 471.
- [37] E. Gallicchio, K. Paris, R. M. Levy, *J. Chem. Theory Comput.* **2009**, 5, 2544.
- [38] V. Barone, M. Cossi, J. Tomasi, *J. Chem. Phys.* **1997**, 107, 3210.
- [39] E. Gallicchio, R. M. Levy, *J. Comput. Chem.* **2004**, 25, 479.
- [40] J. D. Weeks, D. Chandler, H. C. Andersen, *J. Chem. Phys.* **1971**, 54, 5237.
- [41] D. Eisenberg, A. D. McLachlan, *Nature* **1986**, 319, 199.
- [42] E. Gallicchio, L. Y. Zhang, R. M. Levy, *J. Comput. Chem.* **2002**, 23, 517.
- [43] D. Sitkoff, K. A. Sharp, B. Honig, *J. Phys. Chem.* **1994**, 98, 1978.
- [44] J. Wang, W. Wang, S. Huo, M. Lee, P. A. Kollman, *J. Phys. Chem. B* **2001**, 105, 5055.
- [45] X. Pang, H. X. Zhou, *Commun. Comput. Phys.* **2013**, 13, 1.
- [46] A. Chakravorty, Z. Jia, Y. Peng, N. Tajdelyato, L. Wang, E. Alexov, *Front. Mol. Biosci.* **2018**, 5, DOI: 10.3389/fmolb.2018.00025.
- [47] L. Li, C. Li, Z. Zhang, E. Alexov, *J. Chem. Theory Comput.* **2013**, 9, 2126.
- [48] J. Weiser, P. S. Shenkin, W. C. Still, *Biopolymers* **1999**, 50, 373.
- [49] J. A. Grant, B. T. Pickup, *J. Phys. Chem.* **1995**, 99, 3503.
- [50] F. Cazals, H. Kanhere, S. Lorient, *ACM Trans. Math. Softw.* **2011**, 38, 1.
- [51] S. Decherchi, J. Colmenares, C. E. Catalano, M. Spagnuolo, E. Alexov, W. Rocchia, *Commun. Comput. Phys.* **2013**, 13, 61.
- [52] F. M. Richards, *Annu. Rev. Biophys. Bioeng.* **1977**, 6, 151.
- [53] L. Willard, A. Ranjan, H. Zhang, H. Monzavi, R. F. Boyko, B. D. Sykes, D. S. Wishart, *Nucleic Acids Res.* **2003**, 31, 3316.
- [54] G. Kleywegt, T. A. Jones, *Acta Crystallogr. Sect. D Biol. Crystallogr.* **1994**, 50, 178.
- [55] C. R. Chen, G. I. Makhatadze, *BMC Bioinf.* **2015**, 16, DOI: 10.1186/s12859-015-0531-2.
- [56] W. Mattson, B. M. Rice, *Comput. Phys. Commun.* **1999**, 119, 135.
- [57] B. Hess, C. Kutzner, D. van der Spoel, E. Lindahl, *J. Chem. Theory Comput.* **2008**, 4, 435.
- [58] L. Verlet, *Phys. Rev.* **1967**, 159, 98.
- [59] W.-Q. Li, T. Ying, W. Jian, D.-J. Yu, *Comput. Phys. Commun.* **2010**, 181, 1682.
- [60] S. Páll, B. Hess, *Comput. Phys. Commun.* **2013**, 184, 2641.
- [61] V. Yip, R. Elber, *J. Comput. Chem.* **1989**, 10, 921.
- [62] A. Nicholls, B. Honig, *J. Comput. Chem.* **1991**, 12, 435.
- [63] A. Chakravorty, Z. Jia, L. Li, S. Zhao, E. Alexov, *J. Chem. Theory Comput.* **2018**, 14, 1020.
- [64] N. R. Voss, M. Gerstein, *Nucleic Acids Res.* **2010**, 38, W555.
- [65] S. Mitternacht, *F1000Research* **2016**, 5, 189.
- [66] S. J. Hubbard, J. M. Thornton, *Naccess. Computer Program; Department of Biochemistry and Molecular Biology, University College: London*, **1993**.
- [67] M. Schaefer, M. Karplus, *J. Phys. Chem.* **1996**, 100, 1578.
- [68] T. Ikura, Y. Urakubo, N. Ito, *Chem. Phys.* **2004**, 307, 111.
- [69] J. Weiser, P. S. Shenkin, W. C. Still, *J. Comput. Chem.* **1999**, 20, 688.
- [70] B. Zhang, D. Kilburg, P. Eastman, V. S. Pande, E. Gallicchio, *J. Comput. Chem.* **2017**, 38, 740.

Received: 2 November 2018

Revised: 12 December 2018

Accepted: 6 January 2019

Published online on 30 January 2019